Studies on Design Science Research Model

Eteläaho, Elina

2015 Leppävaara

Laurea University of Applied Sciences
Laurea Leppävaara

Studies on Design Science Research Model

Elina Eteläaho
Information Systems
Master’s Thesis
November 2015

Laurea-ammattikorkeakoulu **Tiivistelmä**
Laurea Leppävaara
Tietojärjestelmäosaamisen ylempi AMK-tutkinto

Elina Eteläaho

Studies on Design Science Research Model

Vuosi 2015 Sivumäärä 49

Tässä opinnäytetyössä kehitetään Peffersin ym. (2008) luomaa tietojärjestelmätutkimuksen kehikkoa (design science research model) vastaamaan paremmin tietojärjestelmien kehittämisen tarpeita. Opinnäytetyö on osa tutkimusprosessia, jossa kirjoitettiin artikkeli tieteelliseen konferenssiin (American Conference on Information Systems, AMCIS 2015). Artikkelin tutkimuskysymys on "miten tietojärjestelmätutkimuksen kehikkoa voidaan parantaa tietojärjestelmien kehittämisen näkökulmasta?". Kehikkoon kuuluu kuusi vaihetta: ongelman tunnistaminen, ratkaisun tavoitteiden määrittely, artifaktin suunnittelu ja kehittäminen, ratkaisun esittely, arviointi ja ratkaisun jakaminen.

Tutkimusmetodologiaksi valittiin suunnittelun tutkimus, koska se sopi parhaiten kehikon ja sen käyttömahdollisuuksien arviointiin tietojärjestelmien kehittämisessä. Tutkimus alkoi kirjallisuustutkimuksella, jonka tavoitteena oli hahmottaa keskeiset käsitteet ja selvittää, miten tutkimuskehikkoa oli aiemmin tutkittu ja hyödynnetty. YAMK-opiskelijoiden kurssilla tehdystä ryhmätyöstä oli myös apua kehikon ja sen vaiheiden sisällön hahmottamisessa. Tutkimus jatkui selvittämällä, miten kehikkoa voitaisiin hyödyntää valtion organisaation projekteiden tutkimisessa tieteellisin keinoin. Projektien tutkimukseen käytettiin tapaustutkimuksen menetelmiä, jotka antavat tutkimukselle selkeän rakenteen. Tapaustutkimus toteutettiin pääasiassa teoriaa ja projektien dokumentteja sekä kirjoittajan muistiinpanoja tutkimalla. Dokumentit valittiin huolella, jotta tapaustutkimukselle tärkeä triangulaatio toteutuisi. Dokumentit olivat monen eri henkilön kirjoittamia, ja käsittelivät monipuolisesti erilaisia projekteja, jotka oli toteutettu eri asiakkaille. Näin pyrittiin varmistamaan tutkimuksen luotettavuus. Tapaustutkimuksen toteuttamisen jälkeen tuloksia vertailtiin toiseen julkiseen organisaatioon, jossa tuloksia keräsi toinen tutkija.

Tutkimuksen päätulokset ovat, että käyttäjäkokemuksen ja käytettävyyden pitäisi olla kehikossa näkyvämpiä, ja että tietojärjestelmien kehittämisen näkökulmasta käyttöönotto pitäisi lisätä kehikkoon. Käyttäjiä ja käytettävyyttä ei voida sivuuttaa tietojärjestelmien kehitystyössä. Käyttäjän koulutus järjestelmän käyttöön on myös olennaisen tärkeää järjestelmän käyttöönoton onnistumisen kannalta. Käyttäjän luottamus järjestelmään on helpoimmin rakennettavissa jo projektin aikana, joten käyttäjä kannattaa ottaa mukaan projektiin jo aikaisessa vaiheessa. Vaikka tutkimuksen näkökulmasta käyttöönotto ei olisikaan merkittävä asia, tietojärjestelmien näkökulmasta uusi ratkaisu on todistanut arvonsa vasta onnistuneen käyttöönoton jälkeen.

Tutkimustulokset tuovat myös käytännön hyötyjä: parannetun kehikon käyttö mahdollistaa parempien tietojärjestelmien kehittämisen, koska se huomioi käytännön työssä keskeiset asiat paremmin kuin alkuperäinen vain tutkimukseen keskittynyt kehikko. Lisäksi käytännön työn tutkimisesta voidaan kehittää uusia teorioita.

Asiasanat: tietojärjestelmätutkimus, suunnittelututkimus, tietojärjestelmät, käyttäjäkokemus, käytettävyys

Laurea University of Applied Sciences
Laurea Leppävaara
Information Systems (Master's degree studies)

Abstract

Elina Eteläaho

Studies on Design Science Research Model

Year 2015 Pages 49

This thesis contributes to the development of the design science research model (DSRM) developed by Peffers et al. (2008). This thesis describes the process of writing an article for a scientific conference, the American Conference on Information Systems (AMCIS) 2015. The research question for the article was "how can the design science research model (DSRM) be improved when looking from a systems development perspective?" The framework comprises six phases: identifying the problem; defining the objectives of a solution; designing and developing the artifact; demonstration; evaluation; and communication.

The research methodology chosen was design science research, since it was best suited for evaluating the DSRM framework and its use in the systems development process. The research started with **a** literature review in order to get an understanding of what design science research is, what the DSRM framework is, and what kind of studies had been implemented. Group work done in a Master's degree course also offered valuable information on what is involved in implementing the different phases of the DSRM framework. Research was continued in a public organization, where the task was to find how the framework could be used to examine projects and their progress through scientific methods. The chosen projects (n=10) in the organization were studied with case study methods, offering rigor for the research. The case study was mainly conducted by accessing available documents (n=40) and the notes of the author. The documents used were chosen to offer variety and triangulation; written by different people, describing diverse projects implemented for different clients. Choosing different projects and research methods offers possibility for triangulation thus improving the reliability and validity of the study. After completing the case study, results were compared with results from another public organization studied by another investigator, and an article was written based on mutual conclusions.

The main results are that user experience and usability should be more visible in the DSRM framework throughout the whole process, and that for systems development purposes deployment should be added to the model. User experience and usability have become the key issues in information systems over the years, and neglecting them could have serious consequences. The user should also be trained to use the system in order to make the deployment phase succeed. This is easiest when the intended user trusts the system, and trust can be built with user participation in projects. Even if deployment has less meaning when looking at the framework from a research point of view, only after a successful deployment does the solution prove its value in practice.

These results also benefit practice, enabling the development of better information systems by offering tools for systems development projects in the form of a revised framework that is more suitable for systems development than the more research-oriented framework.

Keywords: design science research, design science, information systems, user experience, usability

Contents

Acknowledgements

The starting point for this study that has now become a Master's thesis was a course held at Laurea University of Applied Sciences. A group of Master's degree students was given the task of thinking about the different phases of the design science research model (DSRM) by Peffers, Tuunanen, Rothenberger and Chatterjee (2008), and describing what was to be included in those phases. The next step was to describe projects experienced at work in light of the model. With a student peer research of what was missing in the DSRM model was conducted, based on the discussion in class and projects in workplaces. The task started by examining articles published on the topic and analyzing the discussion from the class, as well as using work experience in different projects. What started as a course assignment ended as an article accepted to the American Conference on Information Systems (AMCIS 2015) that was held in Puerto Rico in August 2015. The article, titled 'Improvement suggestions for DSRM model', was also published after the conference, even though we were not able to represent it at the conference due to financial reasons. This thesis describes the contents of the article and the process of writing it.

When I started studying in Information Systems Master's program, I did not know how much I would learn during the next two years. Not just by studying, but getting to know new people and understanding my work in a whole new way. Getting new lifelong friendships.

For achieving my goals in getting a second Master's degree I want to thank some people. First, our principal teacher Rauno Pirinen, who has encouraged all of us also to write conference papers, making us understand how theory and practice can be brought together by theorizing about what has been learnt at work, and using theories at work.

My wonderful student colleagues with whom we have had many good discussions, done teamwork in courses and helped each other wherever we can. We have also become amazing friends, and even after some of us have already finished their studies, we still keep in touch. And Pilvi Tuomi, we made a great team writing the article.

And greatest thanks go to my family: Marko, and my parents and siblings. You have seen me tired and frustrated when studying and working simultaneously has been almost too much, but also happy and confident, when I have succeeded. Thank you for your love and support.

Abbreviations

AVI = Regional State Administrative Agency

CIS = Consumer Information Systems

DS/DR/DSR = Design Science / Design Research / Design Science Research

DSRM = Design Science Research Model (Peffers et al. 2008)

ELY = Centre for Economic Development, Transport and the Environment

Evira = Finnish Food Safety Authority

IS/ISR = Information Systems (discipline), Information Systems Research

ICT = Information and Communication Technology

IT = Information Technology

ITIL = Information Technology Infrastructure Library

Mavi = Finnish Agency for Rural Affairs Mavi

MMM = The Ministry of Agriculture and Forestry

RKTL = Finnish Game and Fisheries Research Centre

SOA = Service-oriented architecture

Tike = The Information Centre of the Ministry of Agriculture and Forestry (was merged with National Land Survey in 2015)

1 Introduction

Design Science Research (DSR) is one subcategory of the information systems (IS) discipline, and its core is the research of IT artifacts (see e.g. Orlikowski and Iacono 2001). It is an interesting category, since it focuses on the development of actual artifacts, be they constructs, models or whole systems (for categorization, see e.g. March and Smith 1995; Hevner, March, Park and Ram 2004). Therefore the results of the research may not always be visible to end users, but are all the more important to developers, offering them new development tools. The core of DSR is adding knowledge to the knowledge base, either in form of artifacts or theory. However, the interesting question is: is it possible to utilize DSR also in improving systems development process? In this thesis the topic is examined by using the design science research model (DSRM) by Peffers, Tuunanen, Rothenberger and Chatterjee (2008) and determining whether it could be utilized in systems development process.

In this thesis the practical contribution is based on work experience in different kinds of system development projects. The work environment is a government organization that develops information systems for client organizations and also offers common system parts for organizations in the branch of Ministry of Agriculture and Forestry. The data collected from those system development projects were used in research for the article and this thesis. Since the practical data are from system development projects, the research reflects the views of Nunamaker, Chen and Purdin’s (1991) multimethodological approach to IS research, in which systems development benefits from observation, experimentation and theory building.

1.1 Design science research model

The study is based on improving the design science research model (DSRM) framework, by Peffers et al. (2008). Here is a short description of the model. The DSRM model includes principles of systems development, a practical research methodology and guidance in using it. The method is based on existing research sources and is consistent with the literature. It offers a mental model on which design research can be built, as well as evaluation and representation of the results. DSRM contains six phases that cover the parts of the system development process. (Peffers et al. 2008.) The original framework is included in this thesis as appendix 1.

The development of the system begins with identifying the problem. The problem is something that needs to be solved by developing a new information system or changing an existing system. Problems usually arise for several reasons, which in this study have been divided into six subcategories based on the information gathered from the Master’s study unit. One category of problems is the needs arising from corporate strategy: systems development activities need to follow the path of the strategy. Another reason for creating a new information system or improving an old system is the need to have more efficient and effective systems. Often this need for greater effectiveness is related to a need to reduce employment expenses, or improve data quality. (Ryan and Harrison 2000; Devaraj and Kohli 2002, 6-8, 12).

The third category is the demands arising from the corporate environment, such as amendments or changes in regulations that make an old system useless. The fourth category is risk management or risk minimization. For example, information systems can contain breaks in information security that can over time prove to be very risky. The fifth category is the usability of the system. This is partly a question of efficiency (Maguire 2001), but can be seen as a separate matter because of its importance. Issues associated with bad usability can take a large share of a worker's working hours every day, and systems that don't interact with each other (i.e. so-called legacy systems) may require a great deal of manual work to move information from one system to another (Dedeke 2012). The sixth reason for developing an information system would be innovation to bring competitive advantage in the market (see e.g. Rose et al. 2004). Competition is hard and every firm wants to be the best in its own segment. Creating something that is missing from other firms' offerings can bring in a great cash flow.

In the second phase of the model the objectives of a solution are defined. This phase is based on examining the current situation and understanding it thoroughly. In this phase traditional research methods, such as interviews, observation and measurement, can be quite useful. After gaining an understanding of the current state of the system, the rigorous designing and modeling of the new, improved artifact can begin. In this phase of DSRM the preconditions for budget and schedule are taken into account. Integration into other existing systems and system requirements are described, the data flows from one system to another are modelled, and central constructs are created.

In the third phase the artifact is designed and developed. To do this efficiently, it is imperative that the preceding phase is implemented carefully. The third phase concentrates on weighing the options: what are the possibilities and what would best apply to this purpose? Mapping the possibilities requires a lot of information retrieval. The basis consists of scientific publications, literature, and knowledge of and experience with corresponding information systems. Nunamaker et al. (1991) recommend designing a couple of options and then choosing the best one. Based on the best option a system or a prototype for testing the operability is designed. Nunamaker et al. (1991) and Walls, Widmeyer and El Sawy (1992) emphasize the significance of the theoretical foundation in design science research.

The fourth phase, demonstration, represents the functionality of the information system. In this phase the users can try the new artifact. Also the performance, integration with other systems, and usefulness of the new artifact may be evaluated, and emerging problems fixed. In the fifth phase the feasibility of the solution compared to the original goal is evaluated (see also Hevner et al. 2004). The functionality of the old and new systems is evaluated to determine whether the solution resolves the defined problem. The evaluation phase also offers the chance to test for mapping information security, integration, and usability. In the sixth phase, communication, the contributions of the study are shared with the public in different publications. This is important because only after the new knowledge is out can others utilize it as well (Peffers et al. 2008).

1.2 Systems development in Finnish agricultural administration

The operative environment in my work developing ICT for the Finnish agricultural administration is a complex combination of information systems. Therefore the working information systems and their integration are very important. The organization examined in this study is the Information Centre of the Ministry of Agriculture and Forestry (Tike) that was merged into the Finnish National Land Survey in the beginning of 2015. The projects used for this study are Tike-based, and thus only involve the information systems maintained and developed for, and the platforms offered to, clients in the Finnish agricultural administration that already existed in 2014. The biggest clients are the Finnish Agency for Rural Affairs (Mavi), Finnish Food Safety Authority (Evira), Finnish Forestry Centre, Ministry of Agriculture and Forestry (MMM) and Finnish Game and Fisheries Research Centre (RKTL). Some of the systems are also used in Regional State Administrative Agencies (AVIs), Centres for Economic Development, Transport and the Environment (ELYs), and municipalities, and in some cases even by Finnish farmers. At the moment there are almost 100 working information systems that are being developed, maintained, integrated and monitored. (Tike 2014; MMM 2014.)

As Erol, Sauser, and Boardman (2009) suggest in their article, Tike uses Service Oriented Architecture (SOA) and has been able to create a unified way of system integration for national-level Finnish agriculture and forestry information systems. Tike has quite a unique setting for system development and integration as it is the operative authority in system development for the branch of the Ministry of Agriculture and Forestry. Therefore Tike has been able to tailor the systems and their integration to meet client requirements in the best possible way. The work done in Tike is based on Information Technology Infrastructure Library (ITIL) and best practices that have been collected to form handbooks on different aspects of systems development. Tike has compiled a handbook for system development that is used in every project. The handbook sets the guidelines for projects, whether they are implemented by “traditional” or agile methods. However, Tike also needs to handle some legacy systems of clients that are not based on SOA, so Tike is able to handle some exceptions as well.

Systems development is now mainly based on agile methods, the main components coming from Scrum methods. Project teams are assembled from experts who combine knowledge of IT systems and of the field in which the system in development will be used. Agile methods enable reacting to changes or problems that occur faster than traditional systems development methods. They also rely on constant cooperation with the client, making effective communication within the team and with the client a necessary tool in mantaining functioning project teams (Kasurinen 2013, 27-28.) Often in projects the client (“payer”) and customer (end-user) are not the same, making user participation and understanding user needs challenging, in many cases. Also issues related to maintenance of the systems while other systems are developed can be challenging, especially since there are shared components that need to function at all times.

DSR can be incorporated into projects whenever there are new problems to be solved, and new means to solve them are developed. There are also issues in system integration as well as methods used in development work that need to be handled. Therefore the solutions found can at times be considered to further the development of DSR as well.

When looking at the DSRM from a systems development perspective, it is obvious that demonstration (e.g. in a laboratory environment) may be enough for solely research purposes. Nevertheless, when considering the suitability of such studies for further use, there are some issues that need to be handled. First, the system environment with its many integrations probably cannot be replicated entirely in a laboratory setting. Second, the user probably cannot understand all the possibilities without seeing the new solution in its actual environment, be that the developer or the end-user. Therefore deployment is an important part of the process, tying all the loose ends together.

1.3 Studies of the DSRM model

This investigation of the DSRM model has been divided into three studies. The original paper sent to the conference in February 2015 is handled as study I. In study II, I examine the evaluations of two anonymous reviewers in detail, and defend the conclusions of the original article. In addition, new evidence is presented to prove the points made, and lessons learned in the review process are discussed. Study III is the final article that was accepted to the AMCIS 2015 conference and published (Eteläaho, Tuomi and Pirinen 2015). In the final chapter some further improvements to the model are also examined.

The target of the study was to determine whether the design science research model (Peffers et al. 2008) was lacking something, and it was found that there were things that could be added to the model, at least when considering the DSRM from a systems development perspective. Also some things were found in the model that should be emphasized more clearly. Therefore we argued that user experience and usability should become a more acknowledged part of DSR, and pushed for an understanding of the fact that researchers of DSR deal with real-life problems, as DSR is essentially an applied science. Thus theory should derive in part from real-life projects, in which actual deployment, of an entire information system or a subsystem, was implemented; lessons may be learned from both successful and unsuccessful solutions. The conclusions of the studies are briefly described in chapter 4, contribution of the study.

1.4 The structure of the thesis

This thesis contains five chapters that are represented in Figure 1. In chapter 1, the concept of the DSRM model (Peffers et al. 2008) is presented. Then the operative environment in which one part of the study was executed is described. A short description of the studies is also given. Chapter 2 is dedicated to a literature review, in which basic concepts of DSR are examined. The advancements of DSR in IS are reviewed, as well as the basics of systems development. Then the concepts of user experience, usability and design-development-deployment-dissemination are explained.

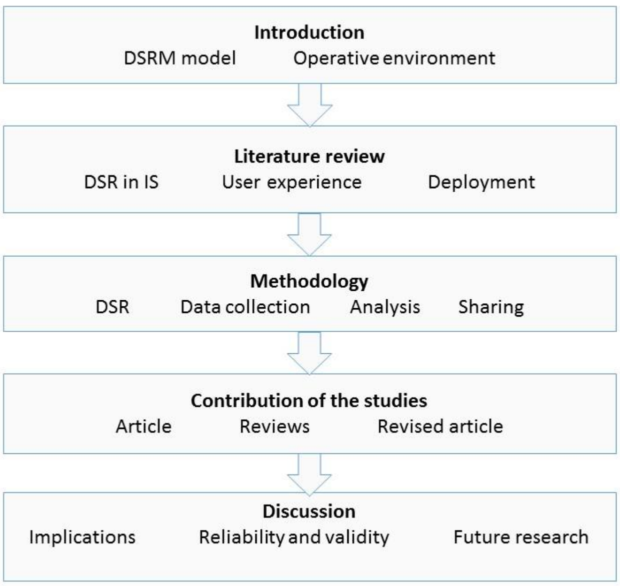


Figure 1. The structure of the thesis.

Chapter 3 concentrates on the methodology used. The research process is described: the collection of data, the analysis and the sharing of the results. In chapter 4 the contribution of the studies is presented. Chapter 5 is discussion, in which the implications of the studies to different fields are examined. Also the reliability and validity of the study are discussed. Finally, some future research possibilities are introduced.

2 Literature review

There is always one or more reasons for developing an IT artifact, be they financial, solving an existing problem, or innovating to achieve the competitive edge. Nevertheless, the reasons usually arise from practice, the real world. When doing research, this should be taken into account. In this literature review, several aspects of DSR in IS are examined through scientific articles and other publications. The practice of building rigorous information systems is described briefly and compared to theoretical perspectives. Then some central concepts such as usability, user experience, and deployment with necessary actions are explained based on literature to create a comprehensive understanding of them.

Haikala and Mikkonen (2011, 25-27) emphasize that there are problems related to academic research on building information systems. The repetition of a project may be impossible, because the people who work on the project may have certain qualities which uniquely influence project outcomes. They also argue that new methods are often deployed before their operability has been academically proven, because the word spreads among practitioners. (Haikala and Mikkonen 2011, 25-28.)

A complete list of reviewed articles (n=28) can be found in the list of research attributes (appendix 2), including many of the central articles of DSR in IS from different angles on the topic. The reviewed works were for example “Systems Development in Information Systems Research” (Nunamaker et al. 1991); “Building an information system design theory for vigilant EIS” (Walls et al. 1992); “Design and Natural Science Research on Information Technology” (March and Smith 1995); “The Sciences of the Artificial” (Simon 1996); “A Design Theory for Systems that Support Emergent Knowledge Processes” (Markus et al. 2002); “Design Science in Information Systems Research” (Hevner et al. 2004); “Anatomy of Design Theory” (Gregor and Jones 2007); and “Theory and Practice of Design Research in Information Systems” (Hevner and Chatterjee 2010).

2.1 DSR in IS discipline

Even if DSR is considered to be a separate part of the IS discipline, the concepts of IS and DSR are intertwined in scientific publications. That is why a short introduction to the IS discipline is necessary to start this chapter. The information systems (IS) discipline examines information systems (artifacts), people, organizations, technology, and interaction (see e.g. Hevner et al. 2004; McLeod and MacDonell 2011). It differs from strictly IT-based disciplines in one relevant issue: it is multidisciplinary. This means that it examines not only technological aspects of information systems like computer science, or social aspects like behavioral sciences, but the intersection of these aspects: the socio-technological phenomena that occur when people and technology are brought together (Baskerville and Myers 2002; Avison and Elliot 2006).

Weber (1987) noticed that information science was not progressing to become an independent discipline, but merely applying theories from other disciplines. However, information science has benefited systems development by enabling learning from past development projects, and by creating unified theories and methodologies (Abdel-Hamid and Madnick 1989). Benbasat and Zmud (2003) argued information science was in a long-term crisis, unable to define the core of the discipline. The argument of Baskerville and Myers (2002) was that IS has emerged as a discipline in its own right and provide a table of knowledge that is unique to IS. They also stated that the value of IS related research is in focusing on socio-technical phenomena, not just behavioral or technological aspects (Baskerville and Myers 2002). Recently Grover and Lyytinen (2015) have expressed their concern of the extensive use of mid-range theories [using reference theories from other disciplines] in IS studies, worried that the research does not reveal the core of IT-related phenomena.

When considering IS from the DSR point of view, it is also noticed that while the IT-artifact is at the core of IS research, it often seems to disappear and must be found again (Orlikowski and Iacono 2001; Hevner et al. 2004). The artifact can be material or abstract, thus being for example a product, database, or model. According to Zhang et al. (2011) there are somewhat differing interpretations of the technological and socio-technological artifact among information science scholars, but in all interpretations, artifacts can be seen to be instrumental and contextual and are applied in organizational and personal settings (Benbasat and Zmud 2003; Orlikowski and Iacono 2001; Hevner et al. 2004). Just like theory and methodology, the artifact itself is not unchangeable, but rather an ever-changing resilient and mutable entity that can be conceptualized from many different perspectives. And even though technologies are at the core they cannot be totally separated from their cultural and path-dependent context (Orlikowski and Iacono 2001; Pirinen 2015). Alter (2003) offers a wider view on the topic, replacing the IT-artifact with the term ‘IT-relient work system’, that also comprises people, risks, costs, and communication towards business, thus making it a different entity.

One of the first articles regarding DSR was from Nunamaker et al. (1991). In the article they introduced the idea of using the systems development process as a research process for DSR. Figure 2 shows how systems development is tied to theory building, observation and experimentation. However, this way of doing research mainly answers the question of whole systems or parts of systems, not the constructs inside systems, or methods. In the article they also note that no single research method alone can offer enough information on a complex topic such as information systems (Nunamaker et al. 1991). Their thoughts emphasize very well an understanding of information system design and development that researchers must consider as well.

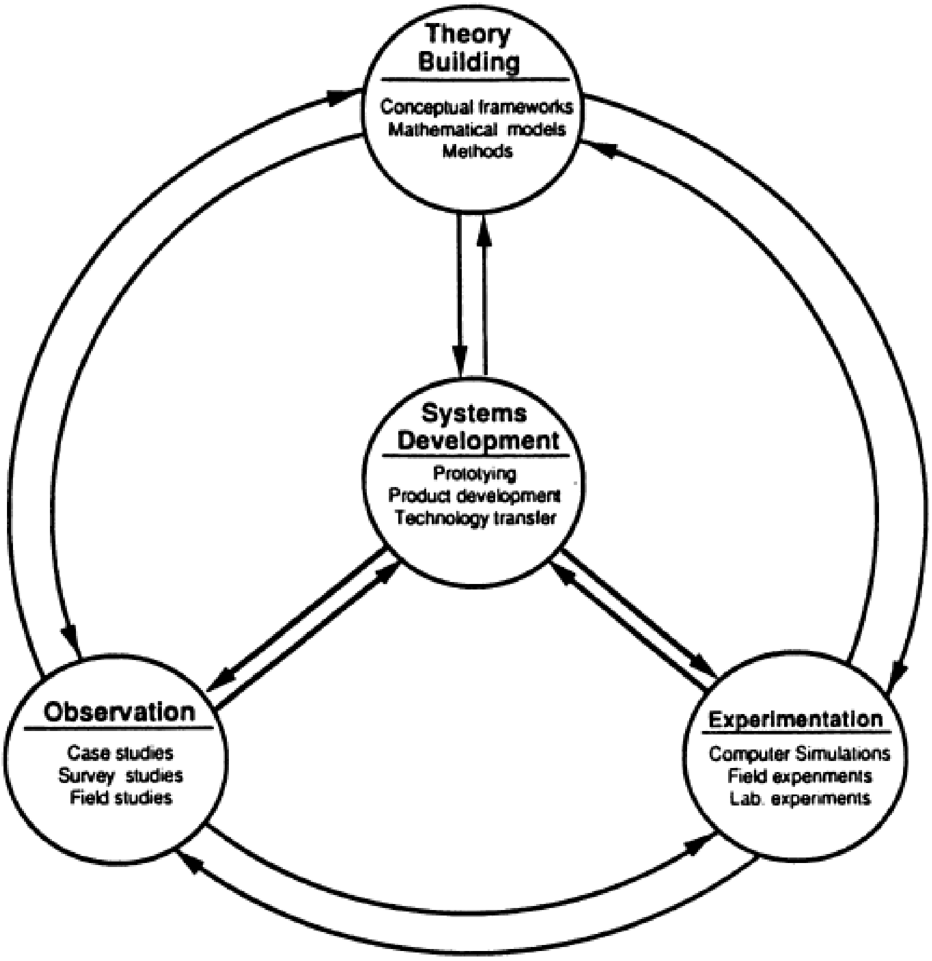


Figure 2. A multimethodological approach to systems development. (Nunamaker et al. 1991).

The second article that is considered very significant in furthering DSR research is from Walls et al. (1992). This article is not related to Nunamaker et al. (1991) but has a more theoretical point of view. Walls et al. (1992) describe DSR as being a prescriptive science, since it tells what is (descriptive), what will be (interpretive) and what should be (normative), combining elements from the natural sciences, social sciences and mathematics. The article contains the first classification of IT artifacts into instantiations and methodologies. Walls et al. (1992) also say that the result of research can be a process or an artifact.

March and Smith (1995) made a classification of IT artifacts into constructs, models, methods and instantiations. While the first three classes are more theoretical, the last contains actual implemented or prototype systems. The two processes are building and evaluation. March and Smith (1995) also discuss the distinction between natural and artificial sciences, finding also that DSR contains elements of both (cf. Walls et al. 1992). In 'The sciences of the artificial' (Simon 1996, earlier version 1969) Simon noticed that after the domination of the natural sciences, the science of design had been emerging since the 1970s after the publication of the first edition of his book, the book itself being one of the drivers. Many of the newer articles have their roots in Simon's thoughts.

In their article Hevner et al. (2004) introduced an information systems research framework which is presented in figure 3. They also provide seven guidelines for design science research. Hevner et al. (2004) also state that technology and behavior cannot be separated in IS. They also argue that the artifact does not exist without the user.

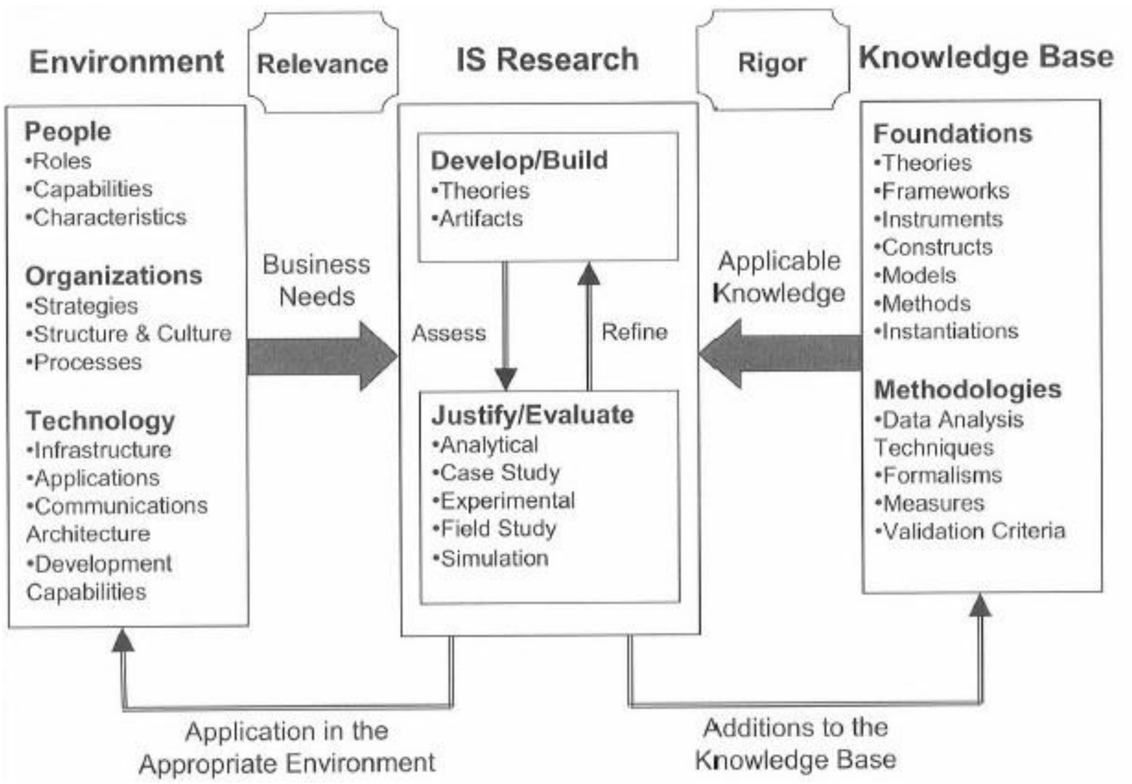


Figure 3. Information Systems Research framework. Source: Hevner et al. 2004.

As Gregor and Hevner (2013) conclude, there are two differing views of DSR. The more theoretical view is the one presented by Walls et al. (1992) and Gregor and Jones (2007). The more pragmatic view is the one represented by Nunamaker (1991), March and Smith (1995) and continued by Hevner et al. (2004) and Hevner and Chatterjee (2010). Peffers et al. (2006,

2008) created the model for DSR based on earlier research. Different views of DSR have also been criticized by e.g. Germonprez et al. (2011), Lyytinen and King (2004) and Iivari (2007).

First, Nunamaker et al. (1991) represented the study of design following the steps of information systems development. This was not enough, since DSR is not just making whole information systems, but also the smaller pieces from which systems are developed, such as databases or ways to handle objects. Walls et al. (1992) brought up the concept of the IT artifact, especially from the design theory point of view. March and Smith (1995) developed DSR further, adding two processes and four artifact types DSR may generate. The article by Hevner et al. (2004) continued the evolution even more in a socio-technical direction.

One of the most difficult issues in DSR is how it can be differentiated from the IS discipline as a whole. It seems that there are only a handful of studies strictly concerning DSR in scientific databases. I ran a search in the biggest databases (EBSCOhost, ProQuest) with the keywords 'design science research', 'design science' + IS, 'design research' + IS, and did not find much. While Walls et al. (2004) had expected there would be 'a rush' for DS research after their article in 1992 (Walls et al. 1992), they found twelve years later that it had not happened. Ten years later Goes (2014) found that after the article published by Hevner et al. in 2004, less than 5 percent of the studies published in MISQ were about design science, five of the studies being published in a special edition in 2008. Goes also concludes that "the main objective is to create knowledge through meaningful solutions that survive rigorous validations through proof of concept, proof of use, and proof of value" (Goes 2014, vi). Therefore theory, the thing that has been found to be insufficiently developed in DSR studies, should not be the only criterion used to evaluate these studies.

Gregor and Hevner's (2013) division of DSR contributions, i.e. the research that could be published, comprises following categories: new solutions for new problems (invention), new solutions for known problems (improvement) and adapting known solutions for new problems (exaptation). Most of the research falls into improvements. Inventions are rare. Exaptation is most likely with individuals who have worked within many disciplines. Nevertheless, it is also possible that routine design can lead to adding to the knowledge base, even if it is rare. The pivotal factor in DSR is adding knowledge to the knowledge base in the form of an artifact or theory development. (Gregor and Hevner 2013.)

Many of these DSR related articles have been criticized by Germonprez et al. (2011). They find that the user should be included in the design process and even claim that the user is also a designer. They call it secondary design, since the user normally has a more specific role after the initial system is developed and released, influencing the development of subsequent versions (Germonprez et al. 2011). Nunamaker and Briggs (2011) are trying to make DSR distinct from other disciplines by highlighting the IT artifact, but the information system and its user cannot be totally separated. That is because every information system needs to offer added value for its user, to fulfill user needs. Thus one focus in designing the systems must be the user experience.

Lyytinen and King (2004, 235) state: “Moreover, it makes no sense for an academic field to focus attention exclusively on any artifact because artifacts never deliver value in their own right. They are complementary assets in production, and their value cannot be understood without the context of their application.” After reading this, even if the authors are referring to the IS discipline specifically (not DSR), it is difficult to understand how the IT artifact could be of any value without users. In the IS discipline the user should be even more important than in DSR, since the focus differs significantly.

In his article Iivari (2007) criticizes some issues in design science. First, there are problems in reducing the knowledge of IT artifacts to theoretical knowledge. Design science also lacks constructive research methods that would bring transparency, rigor and discipline to DSR. One important notion is also the fact that design science is not and cannot be value-free. Because DSR is an applied science, the basic question is: “how can the world be changed?”, and it should produce results that can also be utilized in practice. Thus it cannot be just producing knowledge to understand phenomena like in natural sciences, but it should also solve practical problems. The value of the IT artifact is measured by its usefulness to human purposes. (Iivari 2007.)

There seem to be two very different views of DSR altogether, one being more pragmatic and the other purely theoretical. When we discuss science, the need for theorizing is evident. But there should be more room for innovation, for investigating the practical implications of theory. Research results might then be used not only in academics, but in the real world as well.

2.2 Research for building and improving information systems

Building information systems is a demanding task. The systems have become more complex over the years, and thus building them requires a rigorous approach. This is in line with the ideas of the IS discipline and DSR. Since the reasons for developing information systems are mainly the same that lead to IS research, to better understand the basis for scientific research, the basic principles of building information systems are presented.

There is always a reason for creating a new information system or developing an existing one further. These are usually motivated by enhancing profitability, performance, or customer value, or innovating to gain a competitive edge (Devaraj and Kohli 2002, 6-8). The improvements to old systems or new systems demand a rigorous control process, for example the use of best practices. In many cases the development process is divided into several steps. The first is defining the need for the system, finding out why it is needed. Second is specifying the requirements for the system: functional, non-functional, and business. The requirements are analyzed to define specifications for the necessary databases, interfaces, performance, usability, and security. Then designing can begin, providing technical specifications. Implementation and testing are done next. It is usual that the specifications change during the process, and therefore the different steps overlap, or are even cyclical. Finally, the system is deployed and the maintenance can begin. Typically some changes to the system are made during its lifecycle, and these changes can be the beginning of new projects for improving the system (Pohjonen 2002, 23-39).

One important factor in building information systems today is the participation of the users. The users have become more demanding, and can greatly affect the acceptance of the system. Therefore the design, develop and deploy process needs to be carefully planned. Several studies have discussed the problems faced in IS/IT projects.

Orlikowski (1993) has categorized changes caused by building new informations systems (or improving existing systems) for organizational use. First, categorization is made by the type of change: does the change concern information core, administrative core or technical core? Second, the amount of change is considered: low/moderate change (incremental changes that don't affect work processes) and great degree/total change (radical changes greatly affecting those processes). The greater the changes caused to the employee and working processes are, the greater are the possible problems when considering the acceptance of the system, as well as costs and benefits that may not have been taken into account when planning the project (Orlikowski 1993).

In their study Ryan and Harrison (2000) recognize several intangible benefits and costs that are rarely mentioned by managers as ones that are thought of when making decisions regarding information system projects. The same problems have arisen in the study of McLeod and MacDonell (2011). These are presented in Table 1. As can easily be detected, the technical issues (costs and benefits) are taken much more closely into consideration than "softer" human issues, perhaps because financial decision-makers may not be as aware of these issues as they should be, even if they are told about them. Also technical costs may be easier to estimate moneywise. The biggest surprise was that change management was so neglected. It seems that the changes needed in organizational processes when making a larger IT investment (i.e. building a new information system for an existing process or changing an existing system) are not thoroughly understood.

Table 1 presents intangible costs and benefits associated with IS investment decisions. Even if it describes the decisions from a business angle, it reflects the problems that are common in information systems development processes and especially affect the success of the system deployment phase. Devaraj and Kohli (2002, 17) also highlight the fact that technology and business processes are complementary and IT cannot be developed in isolation. The table shows that even though financial matters are usually taken into account very well, change management, users, and user reactions are not. This can lead to serious problems in acceptance of the system and even end in rejection of the whole project.

Table 1. Intangible costs and benefits and how they are taken into account in IT/IS projects and investment decisions. Source for information: Ryan and Harrison 2000; McLeod and MacDonnell 2011.

Identified as tangible, taken into account	**Identified as intangible, not taken into account in majority of decisions**
training expenses	effects to productivity (learning curve for the new systems, possible delays etc.)
labor savings	quality of work
long-term technical costs and benefits	change management (planning and implementation)
duration that the system will be used	increased role conflict
anticipated depreciation	greater employee empowerment
future maintenance	better communication
costs of updates and upgrades	impact of IT on employees' feeling of loss of power or control
expandability	HR policy changes (in radical changes)
	resistance of new systems (radical changes that change the process etc.) -> acceptance of the new systems is vital for the project to be beneficial

According to Pentland and Feldman (2008), one problem in IS projects is also that the new or improved system does not result in changing the patterns of action. It is easy to design an artifact, but difficult to change associated organizational routines. Even if the solution is accepted in theory, it may not affect the ways people use the system; they may use detours to avoid the new system, or use the new system only partially. It is also possible that issues with trust in other user groups of the system may affect user behavior. Pentland and Feldman (2008) identify two ways of approaching the topic. The materialist view is close to technological determinism. This view argues that technology and its built-in structures enable and constrain certain kinds of behavior. The other, agency view, is closer to social determinism. Its argument is that people can quite freely choose the ways in which they use technology (Berg 1998 according to Pentland and Feldman 2008).

2.3 Focus in user experience and usability

Users are a debated part of the design process. There are different types of users of information systems. First, the operational users that use the system for their daily work tasks. Second, the supervisory users who do not use the system as often but have a local understanding of the system and might be the connection between the developers and operational users, which may not be a good thing. The executive users may not know specifically what the system does, but know its strategic importance and place from the firm's perspective (Pohjonen 2002, 47-48). Users also have different kinds of needs for the system, and therefore systems cannot be tailored for only one specified user group (Nielsen 1993, 23).

According to Nielsen (1993, 24-26) usability is part of the system's practical acceptability. Practical acceptability means that the system needs to be for example useful, reliable and compatible with other systems. Usability is part of the system's usefulness. By definition usability consists of learnability, efficiency, memorability and satisfaction, and minimizes the possibility of user errors. Due to different user needs usability is always a set of compromises. Norman (1988) has also described important aspects of usability in his book "The psychology of everyday things" that is one of the first books about usability. Norman was also the first to use the term user-centered design that was based on seven principles. First, you need to use knowledge in the world and knowledge in the user's head. Second, the tasks need to be simple enough to not overload the user's short-term and long-term memories. Third, things need to be visible so that execution and evaluation can be understandable. Fourth, the mapping needs to be clear. Fifth, there need to be natural and artificial constraints so the user only seems to have one thing to do. Sixth, the user needs the recovery option in case s/he makes an error. Seventh, standardization is the last option, when all else fails (Norman 1988).

According to ISO standard 25010 (ISO/IEC 25010 2011) that covers issues related to software quality requirements and evaluation, usability can be divided into appropriateness recognizability, learnability, operability, user error protection, user interface aesthetics and accessibility. Of these the appropriateness recognizability is particularly interesting, since it emphasizes the fact that the user needs to be able to understand the suitability of the system for the intended use. Standards ISO 9241-11 and ISO 9241-210 focus more on usability and human-centered design. Basic concepts from the standards are described in figure 4. In ISO standard 13407 the user-centered design process is described (Väänänen-Vainio-Mattila 2011, 108), and the process is quite similar to the DSRM process, even though the user is in a more central position.

Some basic concepts of usability and user experience

- **User** is a person interacting with the product
- **Usability** is the "extent to which a system, product or service can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use"
  - Usability criteria can be used to assess aspects of user experience
- **User experience** is "person's perceptions and responses resulting from the user and/or anticipated use of a product, system or service"
  - Includes e.g. emotions, beliefs, preferences, perceptions, responses occurring before, during and after use
  - Consequence of several internal (e.g. attitudes, skills) and external factors (e.g. brand image, functionality), also hardware and software
- **Human-centred design** is an "approach to systems design and development that aims to make interactive systems more usable by focusing on the user of the system and applying human factors/ergonomics and usability knowledge and techniques"

Sources: SFS-EN-ISO 9241-11; SFS-EN-ISO 9241-210

Why human-centred design?

- Increasing productivity and efficiency
- Ease of use: reduced training and support costs
- Increase accessibility
- Improve user experience
- Competitive advantage; sustainability

Principles:

- Explicit understanding of users (includes also stakeholder groups), tasks and environments
- Users involved throughout design and development – understanding the context of use and how the work is done. Active involvement is crucial.
- Iterative process
- Designing the whole user experience, considering also e.g. organizational impacts, user documentation, support and maintenance
- Design team with multidisciplinary skills and perspectives

Figure 4. Basic concepts of usability and user experience and principles of human/user-centered design.

Users also have different levels of understanding of IT and how the systems work from the technical perspective. "Amateurs" may only be interested in the system doing the tasks that they need in order to do their work, whereas experts may understand the technical details very well. When considering the usability of the system, the different starting points of the users need to be understood (Pohjonen 2002, 48). Problems associated with bad usability are underuse, where the user does not understand all the possibilities technical solutions offer, and inaccurate allocation of technology, where solutions are built for non-existent needs. The rapid development and increasing complexity of technology make taking the user into account even more important. If usability is not taken into consideration, the systems built may not be used at all and thus there is no point in developing new systems (Saariluoma 2011, 45-47).

User experience is a more complicated concept. Davis (2003) argues that experience is a process rather than an object. It is intangible and happens between humans and in the world that exists in human minds (Davis 2003). User experience is usually associated with information systems and the interaction between people and those systems. User experience comprises emotions, beliefs, and prior experiences, and thus is different for every user (Davis 2003). In her dissertation, Luojus (2010) states that evolving technology has created a need for finding new concepts and new methods for research of user experience. Therefore she has formulated a concept of ‘expanded user experience’ (eUX) that takes into account both short- and long-term aspects of people’s experience with technology. She concludes that by studying user experience from both short- and long-term aspects, it can be seen as a wider and more meaningful phenomenon (Luojus 2010).

The user and the system are dependent on each other and define each other. Therefore socio-technical research is important. In the Nordic countries participatory design is one way to involve users in the design process (Oulasvirta 2011, 22).

2.4 Design, development and deployment

The design-development-deployment –cycle has been described to some extent in section 2.2 while discussing the information system development process. However, the more theoretical cycle can be described by using the DSRM model of Peffers et al. (2008) with the addition of a deployment phase. After rounds of iterations are made, it is possible to deploy a solution to an actual environment in order determine whether it is suitable for the intended use. Deployment may not be so significant theoretically, but when discussing product or service development, laboratory experiments may not be enough to determine whether a solution solves a given problem, since there are many questions that cannot be answered by focusing on the artifact in isolation. Therefore the solution should be used in practice for a while before final conclusions of its suitability can be made.

In practice, deployment is also a critical point for evaluating the success of an information system project. In this phase all the possible errors made in earlier phases show: do the users accept the product/service; is it good for its purpose; have the efforts made in training and commitment of people been successful; is the management truly committed to the changes made? One important issue is trust in the system (Oulasvirta 2011, 16). If users do not trust the system they will not use it, or will use only some parts of it (see e.g. Pentland and Feldman 2008).

3 Research methodology

The task was to determine whether the model of Peffers et al. (2008) for design science research was missing something, and whether it could be used for researching systems development projects. Therefore the model was investigated thoroughly to understand what its main components are. Additional data were collected from the Master’s study unit called

‘Tietojärjestelmätutkimus’ (Information System Research) and the model was used to evaluate practices in my workplace. To determine what the current situation was, a case study was conducted. Case study is described in more detail in several references (see e.g. Yin 2014; Benbasat, Goldstein and Mead 1987; Gerring 2007; Eisenhardt 1989).

This research consists of the following parts: first, a literature study was conducted, described in more detail in chapter 2. Second, data was collected from the study unit and from practice (i.e. my workplace). Data collection from my workplace was done using case study research, because it would answer the question of how the issues related to the model are seen in current real-world situations. Third, the data were analyzed using qualitative methods, and compared to the DSRM model and results from the study unit. The research followed Yin’s six phase model (see Yin 2014). The model is represented in figure 5.

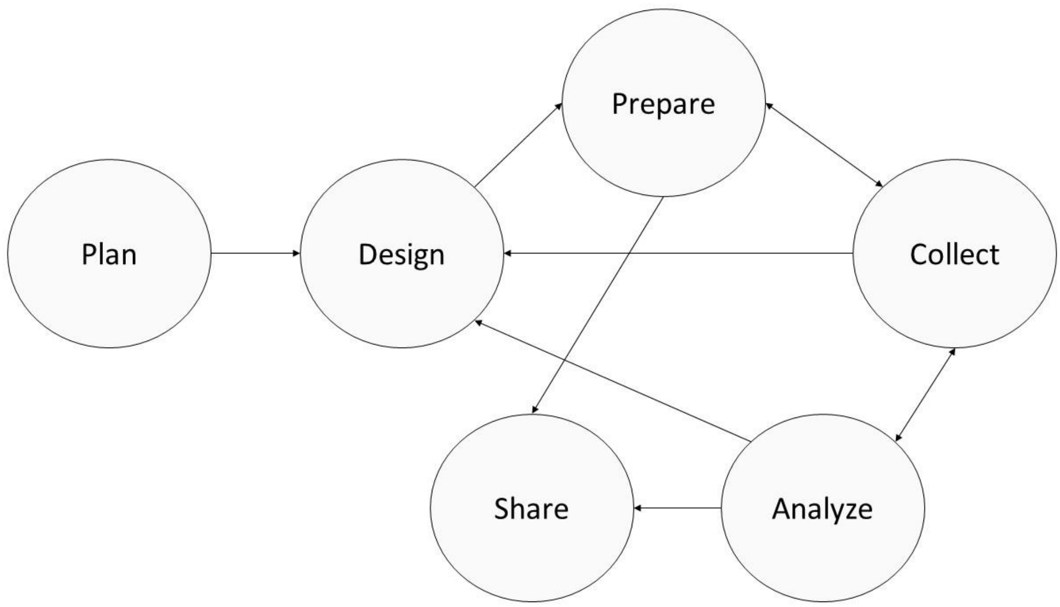


Figure 5. Research process. Source: Yin 2014, 1.

3.1 DSR as research approach

Gregor (2006) discusses the topic of whether or not there is need for theory in design research. As Walls et al. (1992) and March and Smith (1995) have concluded, there are parts of IS research that are consistent with the natural sciences, explaining ‘what something is’. But Gregor (2006) sees the need for theories of design and action, in order to determine ‘how to do something’. In DSR, the scope is both the design product and design process (see e.g. Walls et al. 1992).

Gregor and Hevner (2013) have divided DSR contributions, i.e. research that could be published, into the following categories: new solutions for new problems (invention), new solutions for known problems (improvement) and adapting known solutions for new problems (exaptation). Most of the research concerns improvements. Inventions are rare. Exaptations are most likely to be developed by individuals who have worked within many disciplines. Nevertheless, it is also possible that routine design can lead to additions to the knowledge base, even if it is rare. They conclude that “the key differentiator between professional design and DSR is the clear identification of contributions to the Ω and Λ [artifacts and theory] knowledge bases in DSR and the communication of these contributions to the stakeholder communities.” (Gregor and Hevner 2013, 347.)

Hevner et al. (2004) identify seven features of a successfully implemented design research process. First, the artifact must solve the defined problem. It also must solve a problem not solved before. It should be possible to evaluate the benefits, efficiency and value of the artifact with different measures. The research has to be scientifically proven afterwards. When developing the artifact the solution must be found by combining theory and practice, and finally the results need to be published for a suitable audience (Hevner et al. 2004).

People (users) are mentioned by Hevner and Chatterjee (2010), but mainly in the role of consumers for whom the design is made, and as part of the focus groups – not as people developing the information systems. This may not be enough to understand user needs at a deeper level, especially if discussing more specific information systems, not only web pages.

Since the focus of this thesis is on suggesting improvements to the DSRM framework by Peffers et al. (2008), it concentrates mainly on issues related to the framework and systems development.

3.2 Data collection

Collecting data for the study was done by several methods. First, the DSRM model was carefully examined. After that a literature review on IS and especially DSR was made to have a better idea of past and ongoing research. The literature review is presented in chapter 2. Since the goal was to apply DSRM to actual information systems development projects, it was also important to understand the results of studies of success factors or problems faced in information systems projects. Therefore literature review sections from 2.2 to 2.4 deal with issues related to these topics, covering research of information systems development projects, studies and concepts of user experience and usability, and deployment of information systems or new concepts.

Second, I used data collected in the study unit (Information Systems Research) from other Master's degree students. We had a task in which we needed to tell what was to be included in different phases of the Peffers et al. (2008) DSRM model. There were 12 students in the class, and the principal lecturer wrote students' answers on a whiteboard. I wrote down all the answers as well, in order to use them in my study.

Third, I collected data from different information system development projects (n=10) in my workplace, as well as projects in which I have participated over the last 15 years. The data collection was mainly done using case study methods (see e.g. Yin 2014), focusing more on recent projects, but also comparing them with older projects. The data used is mainly project documentation (n=40) and personal notes. Documents used are requirements (technical and business), user research, test reports, project plans, and final reports. The research was conducted using unobtrusive methods (see e.g. Gray 2009, 424-437), largely by reviewing these documents, ensuring documents selected were representative of different projects (i.e. requirement, development, and deployment projects) implemented for different clients, and written by different members of project teams. This provided one possibility for triangulation. The documents and my own experiences have provided a clear picture of what the success factors and problems in development projects are, and how DSRM could be used, related the data to DSRM and other studies made on the topic.

3.3 Analysis and sharing

The analysis of data was done using qualitative methods. According to Denzin and Lincoln (1994, 2) qualitative research is "multimethod in focus, involving an interpretive, naturalistic approach to its subject matter" (see also Miles and Huberman 1994 or Miles, Huberman and Saldaña 2014). The reliability of qualitative research has been questioned in the literature, and the results of such research have been discussed as potentially biased; however, when multiple methods for collecting data are used, qualitative studies have been recognized as a valid form of research. Of course, the results are always interpretations of the data set, but local groundedness, richness and holism can reveal complexity that might otherwise remain invisible. (Miles et al. 2014, 11; Walsham 2006; Silverman 2011.)

The literature provided information on basic DSR concepts, and on what kind of research has already been done in the field. The analysis of the data gathered in the class was done by using the pattern matching technique, trying to find themes that unite different topics (see e.g. Yin 2014, 143-147; Corbin and Strauss 2013, 216-220; Miles et al. 2014). Especially in the first phase of the DSRM, identifying the problem, pattern matching offered better understanding and categorization of the motivations for developing new information systems or making existing systems better. The results with the categorizations are presented together with the details of the DSRM model in section 1.1.

Information from different projects was also analyzed with the pattern matching technique, trying to find success factors and problems that were common in all or most projects. The factors were themed and categorized, in an effort to find the concepts best describing them. The common factors were compared, and comparisons with students' answers from class were made to try and match them. Prior research on systems development project success factors and problems (e.g. Orlikowski 1993; Pentland and Feldman 2008; McLeod and MacDonell 2011; Strong and Volkoff 2010) was also used.

The analysis was partly based on grounded theory building (Corbin and Strauss 2013; Eisenhardt 1989) as well as qualitative case study methods (e.g. Yin 2014; Gerring 2007; Wynn and Williams 2012), and thus the material was collected until a saturation point was reached, at which similar success factors (or problems) had been found in several projects, and no new similarities were likely to be found. The documents from the work projects were related to phases in the DSRM model to see if there were issues that were not considered in the model. Since the projects themselves are hard to replicate, the documents were used for analysis, as they provide a permanent record of each project that any researcher may reference, and are not dependent on personal experience or recollection.

In writing the article, the common issues found in projects from my organization were compared with the issues found in projects from the Finnish Broadcasting Company (Yleisradio), where my coauthor works. This created further opportunity to generalize the results, since the analysis was done comparing issues that arose in two organizations, and in several projects, and were recorded by two different observers.

Even if the model is meant for research purposes, using the model for the evaluation of information systems projects as well would contribute to theory, since some new viewpoints could be found.

3.4 Triangulation of study

In this study, triangulation – the use of multiple sources of evidence - is used. There are four types of triangulation to be found in the literature: data, investigator, theoretical and methodological triangulation. It is considered to be very important for the validity and reliability of a study. (Campbell and Fiske 1959; Miles et al. 2014, 299-303; Robson 2011, 158; Yin 2014, 120-122.) In this study, data triangulation was achieved by using different sources of data, journal articles, and data gathered in the course, and by choosing different kinds of projects for various clients and using their documentation. The data was collected from several projects in two organizations. Investigator triangulation was gained by incorporating contributions from investigators from different organizations, as well as two anonymous reviews of the article. Theoretical triangulation was obtained by gathering different perspectives on the same data set, by searching for scientific studies of DSR in IS in different databases and examining them thoroughly. Methodological triangulation was achieved using several research methods.

3.5 Summary of methodology

DSR as a research approach is useful for studying systems development. The DSR model by Peffers et al. (2008) was used as the reference in evaluating the projects in order to understand if there was something missing from the model. The data collection was done with many methods, using literature of DSR and its development; using the data gathered from the Master's course in information systems and using data from different projects collected using case study methods. The data collection was in many cases done by unobtrusive methods, using 'dead' data, because this data may be considered more objective, not reacting to changes (Gray 2009, 444) or being reliant on people's memory.

The analysis of the collected data was done using mainly inductive reasoning, trying to find patterns in the data (Gray 2009, 14-15). In analysis many cases (projects) were used because patterns needed to be identified to enable categorization of concepts (Gray 2009, 495). It is important to interpret, understand and explain the features found in a data set, to identify special characteristics (Gray 2009, 499-500).

Results were shared by writing a scientific article on DSRM, and also by writing this thesis, in which the research is described in more detail than was possible in the article. Data, investigator, and theoretical triangulation was also achieved.

4 Contribution of the study

In this chapter the contribution of the study is shortly described. This consists of the conclusions made in the first version of the article "Improvement suggestion on DSRM model", the main points raised in reviews of the article, the author's response to those points, and finally important lessons learned in the review process.The chapter concludes with the main points and conclusions of the final version of the article (Eteläaho, Tuomi and Pirinen 2015) that was published in the proceedings of the AMCIS 2015 conference.

4.1 Study I: first version of the article

Eteläaho, Tuomi and Pirinen (2015b) concluded that the DSRM model was lacking certain components, especially those related to instantiations. The viewpoint was mostly from information systems development, because the data were collected mostly from systems development projects. First, the model does not show any user involvement in planning new information systems or in changing existing systems, even when those changes fundamentally alter the system. Second, it is missing the deployment process of the system or part of the system. The model seems to be more concerned about the technical matters concerning the artifact than the users, the basic premise being that the designer knows the purpose for which the system is built, and knows the process and procedures fully, thus making user participation unnecessary. Usability is an essential part of any information system, and user experience should be considered when designing a system. The user should be involved through the whole design process, also making them more committed to the end result.

Many systems fail because technical experts do not fully understand the complicated processes and requirements of the users (Nunamaker and Briggs 2011). Even a badly designed system can function for a long time if it fulfills user needs. With IS methods and models it is possible to learn to design better information systems (Nunamaker and Briggs 2011; Abdel-Hamid and Madnick 1989). However, relying on existing methods may be problematic, as these methods use only what has been done before. The focus should also be on creating something new, and making methods and literature that might be outdated better along the way (Nunamaker and Briggs 2011).

Eteläaho et al. (2015b) also stated, based on examining journal articles related to design science research in IS, and data collected by authors, that the users and deployment of the information system should be more present in research approaches. Therefore user experience is added to the DSRM model (Peffers et al. 2008), which reflects ideas presented by Germonprez et al. (2011), but makes user participation clearer in the model. Also the deployment is given its own box in the model, since certain problems become apparent only after a system is deployed in the intended environment(s).

The conclusions of Eteläaho et al. (2015b) were in short that user experience and deployment of the system are crucial for system development success, and therefore they should be taken into account when doing design research as well. However, these factors should be reflected also in the DSRM model, even if it is a model developed for research purposes. When practical issues are concerned, the deployment of a system or a part of a system must be considered to understand whether the found solution is sufficient for practice as well, not just theory. The changes made in a system should be well managed so the users commit to the changes that may also affect work processes. The right kind of communication is of great importance, as are the means of implementing the change. The managers must also commit to it. The earlier the users participate in the development process, the easier the deployment of the system will be, and the more accepting users will be of the new system and working processes. Figure 6 represents the additions made to the DSRM model.

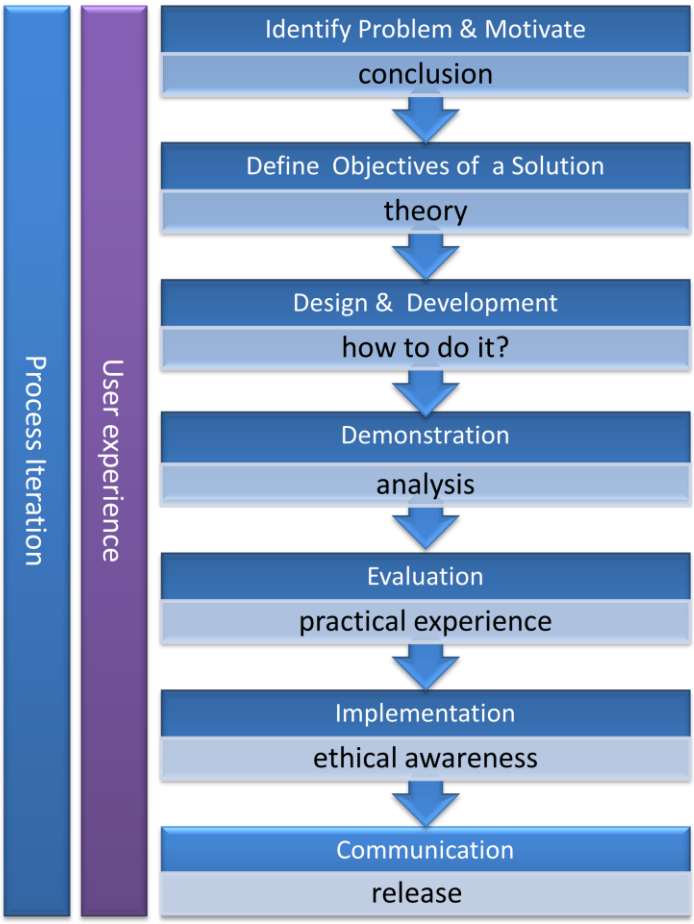


Figure 6. The revised DSRM model. (Eteläaho, Tuomi and Pirinen 2015; revised from Peffers et al. 2008).

Even though the user is not at the heart of the DSRM, and could easily be put aside as part of behavioral research, s/he cannot be forgotten. Therefore the DSRM model should consider the user – at least as a reminder of the reason information systems are developed. There is no point in making information systems or theories for design science research if this essential piece is missing. The deployment of the system is very much tied to the users as well, to their training and to ensuring they use the system in the intended, efficient way.

The contribution of this study (Eteläaho et al. 2015b) is showing that even though some aspects may have been considered while developing the DSRM model, they are missing from the final model and should be emphasized to researchers somehow, if not directly mentioned in the model. An improved version of the model is provided in order to address this issue, especially from the systems development point of view.

4.2 Study II: handling reviews of the article

The reviewers paid particular attention to a few things in the article: first, they found that the article lacked an academic tone. I do admit this may have been the case, since it was the first academic article I participated in writing, not to mention in a foreign language, so the tone and terms were new to me. Second, both reviewers were very keen to claim that the user and usability are embedded in the DSRM. After carefully examining the articles cited, no mention of what was actually meant by those terms could be found. Also user or usability were very rarely mentioned in the articles, even though in practice the users are the reason information systems are built. I understand that DSR is about developing knowledge of IT artifacts and how they are built, but without any possibility of adding that knowledge to real-world implementations, what is the point of the research? At least in applied sciences, which IS and DSR in IS both are, there is a real-life problem to be solved, and if the solution is good enough, it is generalized, both to further theoretical understanding, and to provide a model for developing other information systems. It does not matter whether the solutions are visible to the end user; opportunities to use new methods and models are important for developers, and lead theory forward.

The reviewers also argued that the deployment of the system should not be a part of the process. This prompts the question: how can the results be proved if they are not used in a real-life solution? Especially in the case of instantiations, laboratory testing alone is not enough to prove their suitability (see e.g. Iivari 2007). The deployment process is also tied to change management and trust building (Orlikowski 1993; Pirinen 2013). When new systems are developed and deployed, the management is in a crucial position. No matter how good the new system is, if the users have no trust in the system, the work done is wasted. I do understand the point of the reviewers, but the problem is that they are separating research completely from the real world. To a certain degree that can be done, but in applied sciences seeing research as “an island” is not the right solution. Therefore the process of designing, developing, deploying and disseminating cannot be intercepted.

The first reviewer brought up several other points than those presented in the article, saying they could be used to criticize the DSRM. These points considered everything from the model's novelty to its possible oversimplification and failure to emphasize empirical evaluation. Since the scope of IS is studying socio-technological phenomena (see e.g. Baskerville and Myers 2002), and even if DSR focuses on the IT artifact, our article concluded that the DSRM should also place greater emphasis on user viewpoints. Based on these notions DSR does differ from pure engineering research. All models are simplified, but necessary for research purposes, offering a rigorous approach and tools for research. The objections raised by the reviewer seem dependent on a black and white separation of research and practice. Since my own background is very much in the practice, I do not understand the need for separation. As Gregor and Hevner (2013, 347) state: “the key differentiator between professional design and DSR is the clear identification of contributions to the Ω and Λ [theory and artifacts] knowledge bases in DSR and the communication of these contributions to the stakeholder communities.” This can also be done in a systems development project. Also Nunamaker (1991) argues that systems development with experimentation and observation can also provide research contributions. Therefore it remains unclear what the reviewer means with his/her remarks.

The reviewers used exceptionally bad examples to support their arguments. For example, one reviewer compared the cost of making changes in systems development to the cost of repainting a house. This made no sense, because it is a well-known fact that if you want to repaint a house, you first need to remove the old paint, to prevent the new paint from peeling. Another example claimed that several IT products were all designed without user commitment. As for these examples, it seems that the reviewers are quite estranged from the real world in general or were in a hurry writing their reviews. The arrangers specifically asked the reviewers to be polite, but that was not the case here. My first thought about the reviewers was that they were just trying to knock the article to the ground. As I wrote a review of another article for the conference, and tried to be constructive and polite, the reviews I received made me feel really bad; I expected greater collegiality within the academic community.

The review process was the first for me and as such a very educational experience. I found that the “academic tone” is an important matter, and that one needs to be very careful not to step on anyone’s toes in the academic world. It seems that there is an enormous gap between systems development practice and research, and I think that the real world could teach a thing or two to researchers as well, especially in the context of the applied sciences. Nevertheless, the process taught me a great deal about how to express my ideas in a correct way.

4.3 Study III: changing the article based on the reviews

Based on the reviews the article was reconstructed, especially in means of finding the right kind of tone. The arguments were made clearer and new evidence supporting the arguments was provided. The theoretical foundation was reinforced, and the analysis was explained in more detail. The new suggestion for the DSRM model remained the same, but the second conclusion was modified to answer to the demands of the research world. Findings from the practice of systems development were also emphasized more. The model is useful even as it is, but some enhancements could be made.

4.3.1 User experience and usability to a more visible part of the model

The first conclusion is that user experience and usability are quite invisible in the model. There are indications in the literature that users could be a part of the model, but they remain invisible in the model diagram. It also remains unclear what their actual role in the process is even if they are included in the model. As user needs and user experience have become an increasingly important part of the design process, their role should be emphasized. Therefore, we suggest that user experience should be considered in every part of the process. In an earlier article by Peffers et al. (2006) the role of the users was clear in one of the case studies, but it does not show in the actual model.

There has been a significant change in information systems design in recent years towards more user-friendly systems. User-centered methods in design have become more applicable, and standards concerning usability and user-centered design have emerged (SFS-EN ISO 9241-210 2010; SFS-EN ISO 9241-11 1998; ISO/IEC 25010 2011). This change should be visible also in research frameworks.

One important notion on involving users in the development process (and deployment) is the trust building. Research has shown that if users do not trust a system, they will not use it, or use it only partially (see e.g. Pentland and Feldman 2008). Therefore this aspect should be considered in the design process. Trust building starts from a strategic level, which makes the commitment of management an essential part of it. Also, if the changes in working processes are significant, it is important to train the users and slowly adapt them to the new ways of working.

4.3.2 Adding deployment to the model

The second conclusion is that especially when considering researching system development, deployment should be a part of the model. Theory alone is not enough to prove that solutions work in the real world. Also, only after deploying part or parts of a system is it possible to find whether the solution really works. Many solutions could work but fail because the process is not planned carefully enough. Also the operative environment affects the suitability of the solution, and laboratory testing will not be enough. When thinking strictly from a theoretical point of view, this may not seem so evident, but in practice theory is not sufficient if it cannot be utilized. Thus, since DSR is an applied science, these practical aspects should also be understood.

Deployment is the end of one phase in systems development, even though the use of the system is just beginning. However, the decisions made during the development process affect the deployment phase greatly. As Orlikowski (1993) points out, the magnitude of changes in information systems and how they are handled from the user and management points of view during the project are essential for the deployment phase to be successful. The matters collected in Table 1 show that many things affecting the deployment phase are forgotten or not considered important in earlier stages of the project. If these things are not addressed, the acceptance of the system may be at risk and deployment may fail, leading to the failure of the entire project.

4.4 Summary of the studies

The first version of the article was written more from a practical point of view, and it did not emphasize the theoretical points enough. It also failed to describe the research process in a way that is needed in academic research. The reviews helped me to understand what the article was lacking, even if in a rather rude manner, and to use the right kind of tone in argumentation. The conclusions remained the same but they were represented in a more academic manner.

In the final version of the article the text was rearranged to emphasize those parts that were underrepresented in the original: theory, methodology, and in particular, discussion. Discussion was present in the original, but was not appropriately highlighted. Even if the viewpoint was practical, the article did contribute to theory and revised the DSRM framework for the needs of practice, actual systems development projects. Even in science, it is important to understand the real world since the point of DSR is to solve real world problems.

5 Discussion

These studies offer only a narrow perspective on DSR and improving DSRM, but since this is a Master's thesis, it was important to narrow down the topic. In this chapter implications of the study are discussed. The reliability, validity and limitations of the study are discussed. Some possible future research topics are introduced.

5.1 Implications

There are several implications that arise from the study results. In practical perspective, in systems development projects problems often arise when the design does not take the user into account. Therefore the findings from the literature and the two organizations imply that paying attention to user experience and deployment as part of the system development process would make projects more successful. These results can also be applied to other environments. This study can affect the future research especially from the user perspective, enhancing the building of user trust in the process of developing IT artifacts.

The theoretical implications are that there is a need for developing a second model that looks beyond experimentation in the laboratory, and takes more practical issues into account. For research purposes it might be enough that the concept is proven in a laboratory environment, but in practice all environments are different. Therefore a more practically oriented research model could be a good idea.

Management and leadership implications show that it is clear that the commitment and actions of the management are essential for the success of systems development projects. Managers make decisions about incorporating users' prespectives in development projects, as well as about organizing training, and can therefore greatly affect the outcomes of projects. It is also notable that the process requires leadership, not just management. These implications can also lead to further societal conclusions.

The implications for the operative environment used in the study are that the results clarify that taking the user and usability better into account when designing new information systems or improving existing ones is vital for the success of the projects. Using DSRM could help in learning more about the projects already implemented, and in using the learned things in new projects more effectively. The projects in which users are taken into the process as co-designers, or at least evaluators, have been more successful. This viewpoint is even more important when users are not only public officers, but farmers or other average citizens. Also the deployment process and user training are crucial parts.

Thinking more widely, in Finland there is an ongoing debate about digitalization. The government wants to digitalize, and so do municipalities. Digitalization is one of the so-called “top projects” that are given extra money in the budget. The problem is that nobody seems to know how to execute digitalization wisely, and the attempts have so far not been very successful. To relate this to the results of this study, it is evident that without taking the users (here all the citizens) into account, these attempts probably will not succeed, at least the way they are intended to, no matter how good the design might be. Since the problems lie mainly in shared systems, not in making every organization their own systems, usability, user experience and deployment are critical issues. The solutions need to be fresh, and design science research would go a long way in achieving this.

All in all, the study strengthens the perception that was already found in earlier studies (e.g. Orlikowski 1993; Germonprez 2011; McLeod and MacDonell 2011; Strong and Volkoff 2010) that user experience and usability, as well as deployment, should be taken more into account in systems development projects. As for research, there should be results that also benefit practice, since DSR is after all an applied science. In this subchapter, some possible outcomes are already described.

5.2 Reliability and validity

As in all qualitative studies, the concepts of reliability and validity need to be discussed (Yin 2014; Miles and Huberman 1994; Denzin and Lincoln 1994; Gerring 2007). For this study to be externally valid, results should be similar in other student groups fulfilling the same task, or all Finnish national-level government organizations, as well as other public organizations (like the Finnish Broadcasting Company). I believe that the matters discussed here are very important, and taken into consideration in all companies and organizations. And if they are not, problems are likely to occur. The same applies to private companies, maybe even more than to public sector organizations due to competition. The internal validity of the study is good too, because the same conclusions are likely to be drawn from the material by other researchers as well.

To confirm the reliability of the study, I have collected the documents and notes. I have also provided the background information of why the study was made: by looking at the literature and comparing it to the results of the lecture task and to the lessons learnt in work environments, a clear research gap was identified. Also the data collection procedures have been presented in narrative form and in research attributes (appendix 2). Even though I have tried to make the (manual) database as complete as it can be, there is one thing lacking. Since one of the organizations studied is my workplace, which I know very well, I do have some knowledge that cannot be wholly incorporated into the database. Therefore, for someone not familiar with the organization, independent research might not result in all the same findings even if all the same documents are used.

The studies are based on theory and experiences in projects in two organizations, both of which are public organizations, not private companies. One acquires information systems that are tailored according to its needs, the other designs and implements systems for its clients. Therefore, even if private firms are not used in the study, the differing roles of the two organizations in information system development allow triangulation. The reliability of the study can be considered good because there are four levels of triangulation: data, investigator, theoretical and methodological (see e.g. Yin 2014; Miles and Huberman 1994).

5.3 Limitations and future research

The results are based on projects from two organizations that are both public organizations. Therefore the results may not apply as such to private sector organizations, but similarities are likely to be found. User experience may even be a more important issue in the private sector, since customers pay for their products directly and thus expect even higher quality and usability. In the public sector payment is mostly in the form of taxes, and there may be no choice other than those information systems offered by the authorities. When considering the deployment process, in the private sector mistakes can even cause the whole information system to be taken out of the market, because users have no trust in it.

One limitation of the study is also that the viewpoint is more in practice than in the academic world; however, theories and models should also be used outside the academic world, outside laboratories, for them to prove their value (see Goes 2014), so this study offers a new viewpoint on the topic.

It can also be difficult to simplify and generalize research findings on tailored information systems, because every solution is made based on current needs, and works in a particular system environment. In ERP's or banking systems, solutions are more similar, and more generalizations may be possible than in tailored information systems.

One interesting viewpoint would also be services. Today information systems are complex and increasingly service-oriented, making the IS market more about services, not just products or artifacts. Users also demand more in terms of quality and usability, including ease of use, than 30 years ago. The systems need to create additional value to the customer, and the customer needs to be able to affect the service. Since people expect something more than just a product, DSR should think about this more as well. Does the term “IT artifact” describe the service built around the information systems? Does the model take into account the process of creating a service, and the increasingly common practice of co-creating value with the customer? When designing IT-artifacts that are used in the Internet, the design is not all in the hands of the producer, and even the researcher cannot understand the whole design process if he/she does not understand these phenomena around it.

The concept of service design was first used by Shostack (1982) who found that products and services often coexist, such that one cannot exist without the other. Shostack (1982) also argued that information systems are not products or services, but a combination of both. However, the product-centered thinking stayed dominant for a long time, even when discussing information systems. Tuli et al. (2007) discovered that the customer sees not just the bundle of products a firm offers, but also the requirements as well as the deployment and postdeployment services as a part of the firm's (network) offering. Thus the solutions should be seen as relational processes, not just as a bundle of products and services, and the customer should be able to participate in the process. This agrees very well with the ideas of Shostack (1982) and service-dominant logic – including the customer in the process of value co-creation (Lusch, Vargo and Tanniru 2010) and changing from a product-centered view to the service/solution-centered view. Lamb and Kling (2003) also share the understanding of users as actors: users not only use a system or service, but co-create value, as active agents. These ideas of service design are important when trying to understand the needs of information system users as well. After all, we rarely discuss only the system, but also the service it provides to its users. These thoughts are also consistent with the thoughts of Vartiainen and Tuunanen (2013), and Tuunanen, Myers and Cassab (2010), who argue that a new form of information systems, Consumer Information Systems (CIS), is emerging. This affects also systems development, as users need to be understood as consumers, who (unlike organizational users) focus more on enjoyment and purchases (Tuunanen el al, 2010). Vartiainen and Tuunanen (2013) use geocaching as an example of CIS.

As for future research, the model could also be examined further from the viewpoint of system integration. I studied system integration from the Integration Readiness Levels (IRLs) perspective in an unpublished case study (Eteläaho 2014), and found that many problems in information systems occur because of the complicated integrations and dependencies between systems. Also outsourcing of the firm's tasks has affected the need for working system interfaces and integration. System integration can be seen as one of the core capabilities of a firm. (see e.g. Hobday, Davies and Prencipe 2005; Rolland and Monteiro 2007.) Therefore it would be important to understand how DSRM could be used in integration design research; are there possibilities for theorizing and finding new solutions?

DSR should also evolve with the changes in IS. Even if Baskerville and Myers (2009) argue that research in IS can be in sync with practitioners' interests or even lead, this does not happen in DSR, or at least is not as visible as in IS. Baskerville and Myers (2009) continue that the IS researchers seem to follow the latest IS fashions and do not ignore what is happening around them. They even suggest that the researchers should be involved in setting the fashions and evaluating them. (Baskerville and Myers 2009.) This would probably go a long way in bringing DSR closer to the practitioners' world, and enhance the chances for publications in scientific journals.

5.4 Towards the future of the DSRM model

As our article (Eteläaho et al. 2015) suggested, the DSRM model is usable as it is, especially for research purposes. It has also been used in other sciences, not just IS research. The articles found suggest that DSR is useful in studying service systems engineering (SSE) and decision support systems (DSS). In SSE, according to Böhmann, Leimeister and Möslein (2014) design science research and the model of Peffers et al. (2008) could be a prominent research approach for developing more knowledge for enabling and supporting the engineering of service systems. There can also be challenges: the complexity of service systems might limit the possibilities for iteration and evaluation. Applying different solutions to the same problem can be difficult as well, as the contexts and problems can change. (Böhmann et al. 2014.) In DSS Miah, Kerr and von Hellens (2014) find that DSR could be used to improve DSS development research. They find that DSR theories could be used to develop more user-centered methods to create a "dynamic DSS artefact that will be tailorable for users' design need" (Miah et al. 2014, 262). They find that the context and understanding of end-user needs (work processes, activities) are an important aspect of system design. (Miah et al. 2014.)

Nevertheless, one issue that needs more attention remains: how the model could be more useful also in systems development. The article (Eteläaho et al. 2015) several important ideas: user experience and usability should be more clearly emphasized; deployment should be incorporated into the model; and trust towards the new solutions or new applications of existing solutions should be built. As the model is meant for research purposes, this is not in focus. But to successfully examine the systems development process with DSR, and theorize about that process, it would make sense to create another model. The most important issue besides those mentioned in the article could be spreading the knowledge to developers, not just the academic community. Also communicating the new knowledge in understandable ways would be very important for management to understand and support the new methods too. The model is in many ways close to traditional systems development processes, which makes it easy to use; however, projects using agile methods, in which project goals, and the solutions required to meet those goals, may change rapidly, present another issue. The good thing in the model is that it can focus on a smaller piece of a system. This kind of isolation makes the use of the model easier when using agile methods.

Tuunanen and Peffers (2011) are developing a new theory of targeted service co-design to become one part of DSR, but their article has not been published yet. It offers an interesting viewpoint of connecting value co-creation (Vargo and Lusch 2004; Lusch et al. 2010) and DSR in information systems (Hevner et al. 2004), making it possible to research services/systems that are co-created with customers, not just existing ones but also possible customers. This could offer an interesting future for the use of the DSRM framework as well.

References

Abdel-Hamid, T. K., and Madnick, S. E. 1989. Lessons learned from modeling the dynamics of software development. Management of Computing, 32(12), 1426-1436.

Alter, S. 2003. 18 reasons why IT-reliant work systems should replace “the IT-artifact” as the core subject matter of the IS field. Communications of the Association for Information Systems, 12, 366-395.

Avison, D., and Elliot, S. 2006. Scoping the discipline of information system. In J. L. King, & K. Lyytinen (Eds.), Information systems: The state of the field (pp. 3-18). Chichester, England: John Wiley & Sons.

Baskerville, R. L., and Myers, M. D. 2002. Information systems as a reference discipline. MIS abdelQuarterly, 26(1), 1-14.

Baskerville, R. L., and Myers, M. D. 2009. Fashion waves in information systems research and practice. MIS Quarterly, 33(4), 647-662.

Benbasat, I., Goldstein, D. K., and Mead, M. 1987. The case research strategy in studies of information systems. MIS Quarterly, 11(3), 369-386.

Benbasat, I., and Zmud, R. W. 2003. The identity crisis within the IS discipline: Defining and communicating the discipline's core properties. MIS Quarterly, 27(2), 183-194.

Böhmann, T., Leimeister, J. M., and Möslein, K. 2014. Service systems engineering. A field for future information systems research. Business & Information Systems Engineering, (2), 73-79.

Campbell, D. T., and Fiske, D. W. 1959. Convergent and discriminant validation by the multitrait-multimethod matrix. Psychological Bulletin, 56, 81-105.

Corbin, J., and Strauss, A. 2013. Basics of qualitative research. Techniques and procedures for developing grounded theory. 4th ed. United States of America: SAGE Publications, Inc.

Davis, M. 2003. Theoretical foundations for experiental systems design. ACM SIGMM Workshop on Experiential Telepresence ETP '03, 45-52.

Dedeke, A. 2012. Improving legacy-system sustainability: A systematic approach. IT Pro, January/February.

Denzin, N. K., and Lincoln, Y. S. 1994. Introduction: Entering the field of qualitative research. In N. K. Denzin, & Y. S. Lincoln (Eds.), Handbook of qualitative research (pp. 1-18). United States of America: Sage Publications, Inc.

Devaraj, S., and Kohli, R. 2002. The IT payoff: Measuring the business value of information technology investments. United States of America: Prentice-Hall, Inc.

Eisenhardt, K. M. 1989. Building theories from case study research. Academy of Management. the Academy of Management Review, 14(4), 532-550.

Erol, O., Sauser, B. J., and Boardman, J. T. 2009. Creating enterprise flexibility through service oriented architecture. Global Journal of Flexible Systems Management, 10(1), 11-16.

Eteläaho, E., Tuomi, P., and Pirinen, R. 2015. Improvement suggestions for the DSRM model. American Conference for Information Systems (AMCIS) 2015.

Germonprez, M., Hovorka, D., and Gal, U. 2011. Secondary design: A case of behavioral design science research. Jais, 12(10), 662-683.

Gerring, J. 2007. Case study research : Principles and practices. New York: Cambridge University Press.

Goes, P. B. 2014. Editor's comments: Design science research in top information systems journals. MIS Quarterly, 38(1), iii-iv.

Gray, D. E. 2009. Doing research in the real world. 2nd ed. Great Britain: SAGE Publications.

Gregor, S. 2006. The nature of theory in information systems. MIS Quarterly, 30(3), 611-642.

Gregor, S., and Hevner, A. R. 2013. Positioning and presenting design science research for maximum impact. MIS Quarterly, 37(2), 337-355.

Gregor, S., and Jones, D. 2007. The anatomy of a design theory. Journal of the Association for Information Systems, 8(5), 312-335.

Grover, V., and Lyytinen, K. 2015. New state of play in information systems research: The push to the edges. MIS Quarterly, 39(2), 271-296.

Haikala, I., and Mikkonen, T. 2011. Ohjelmistotuotannon käytännöt. 12th ed. Helsinki: Talentum.

Hevner, A. R., and Chatterjee, S. 2010. Design research in information systems. Theory and practice. New York: Springer.

Hevner, A. R., March, S. T., Park, J., and Ram, S. 2004. Design science in information systems research. MIS Quarterly, 28(1), 75-105.

Hobday, M., Davies, A., and Prencipe, A. 2005. Systems integration: A core capability of the modern corporation. Industrial and Corporate Change, 14(6), 1109-1143.

Iivari, J. 2007. A paradigmatic analysis of information systems as a design science. Scandinavian Journal of Information Systems, 19(2), 39-64.

ISO/IEC 25010. 2015. **ISO/IEC 25010:2011(en).** Systems and software engineering - systems and software quality requirements and evaluation (SQuaRE) - system and software quality models. Retrieved from https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en. Accessed 15.9.2015.

Kasurinen, J. P. 2013. Ohjelmistotestauksen käsikirja. Jyväskylä: Docendo.

Lamb, R., and Kling, R. 2003. Reconceptualizing Users as social actors in information systems research. MIS Quarterly 27(2), 197-235.

Luojus, S. 2010. From a momentary experience to a lasting one. The concept of and research on expanded user experience of mobile devices

Lusch, R., Vargo, S. and Tanniru, M. 2010. Service, value networks and learning. J. of the Acad. Mark. Sci. (2010) 38:19-31.

Lyytinen, K., and King, J. L. 2004. Nothing at the center?: Academic legitimacy in the information systems field. Journal of the Association for Information Systems, 5(6), 220-246.

Maguire, M. 2001. Methods to support human-centred design. International Journal of Human-Computer Studies, 55, 587-634.

March, S. T., and Smith, G. F. 1995. Design and natural science research on information technology. Decision Support Systems, 15(4), 251-266.

Markus, M., Majchrzak, A., and Gasser, L. 2002. A design theory for systems that support emergent knowledge processes. MIS Quarterly, 26(3), 179-212.

McLeod, L., and MacDonell, S. G. 2011. Factors that affect software systems development project outcomes: A survey of research. ACM Computing Surveys, 43(4), 24:1-24:56.

Miah, S. J., Kerr, D., and von Hellens, L. 2014. A collective artefact design of decision support systems: Design science research perspective. Information Technology & People, 27(3), 259-279.

Miles, M. B., Huberman, A. M., and Saldaña, J. 2014. Qualitative data analysis. A methods sourcebook (3rd ed.). United States of America: SAGE Publication, Inc.

Miles, M. B., and Huberman, A. M. 1994. Qualitative data analysis : An expanded sourcebook (2nd ed. ed.). Thousand Oaks: Sage.

MMM 2014. The official webpage of the ministry of agriculture and forestry. Retrieved from http://www.mmm.fi/fi/index/etusivu.html. Accessed 15.5.2014.

Nielsen, J. 1993. Usability engineering. Boston (MA): Academic Press.

Norman, D. A. 1988. The psychology of everyday things. New York: Basic Books.

Nunamaker, J. R., and Briggs, R. O. 2011. Toward a broader vision for information systems . ACM Transactions on Management Information Systems, 2(4), 1-12.

Nunamaker, J. R., Chen, M., and Purdin, T. 1991. Systems development in IS research MIS Quarterly, 7(3), 89-106.

Orlikowski, W. J. 1993. CASE Tools as organizational change: Investigating incremental and radical changes in systems development. MIS Quarterly, 9(1), 309-340.

Orlikowski, W. J., and Iacono, C. S. 2001. Research commentary: Desperately seeking the "'IT'" in IT Research A call to theorizing the IT artifact. Information Systems Research, 12(2), 121-134.

Oulasvirta, A. 2011. Mitä on ihmisen ja tietokoneen vuorovaikutus? In A. Oulasvirta (Ed.), Ihmisen ja tietokoneen vuorovaikutus (pp. 13-42). Tallinna: Gaudeamus.

Peffers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V., and Bragge, J. 2006. The design science research process: A model for producing and presenting information systems research. 83-106.

Peffers, K., Tuunanen, T., Rothenberger, M., and Chatterjee, S. 2008. A design science research methodology for information systems research. Journal of Management Information Systems, 24(3), 45-77.

Pentland, B. T., and Feldman, M. S. 2008. Designing routines: On the folly of designing artifacts, while hoping for patterns of action. Information and Organization, 18, 235-250.

Pirinen, R. 2013. Towards realization of research and development in a university of applied sciences. doctoral dissertation, forestry and natural sciences, publications of the university of eastern finland, 108.

Pirinen, R. 2015. Towards continuum-focused methodology for information systems research and development. EDUCON 2015, Tallinn, 167-175.

Pohjonen, R. 2002. Tietojärjestelmien kehittäminen (2nd ed.). Jyväskylä: Docendo.

Robson, C. 2011. Real world research : A resource for users of social research methods in applied settings (3rd ed. ed.). Chichester: Wiley.

Rolland, K. H., and Monteiro, E. 2007. When 'perfect' integration leads to increasing risks: The case of an integrated information system in a global company. In O. Hanseth, & C. Ciborra (Eds.), Risk, complexity and ICT (pp. 97-117). Bodmin, Cornwall, Great Britain: Edward Elgar Publishing Limited.

Rose, J. M., Rose, A. M., and Strand, N. C. 2004. The evaluation of risky information technology investment decisions. Journal of Information Systems, 18(1), 53-66.

Ryan, S., and Harrison, S. 2000. Considering social subsystem costs and benefits in information technology investment decisions: A view from the field on anticipated payoffs. Journal of Management Information Systems, 16(4), 11-40.

Saariluoma, P. 2011. Käyttäjä. In A. Oulasvirta (Ed.), Ihmisen ja tietokoneen vuorovaikutus (pp. 45-61). Tallinna: Gaudeamus.

SFS-EN ISO 9241-11. 1998. Näyttöpäätteillä tehtävän toimistotyön ergonomiset vaatimukset, osa 11: Käytettävyyden määrittely ja arviointi. Ergonomic requirements for office work with visual display terminals (VDTs), part 11: Guidance on usability (ISO 9241-11:1998). Helsinki: Suomen standardoimisliitto SFS.

SFS-EN ISO 9241-210. 2010. Ihmisen ja järjestelmän vuorovaikutuksen ergonomia. osa 210: Vuorovaikutteisten järjestelmien käyttäjäkeskeinen suunnittelu. Ergonomics of human-system interaction, part 210: Humancentred design for interactive systems 2010 fi/en. Helsinki: Suomen Standardisoimisliitto SFS.

Shostack, G. L. 1982. How to design a service? European Journal of Marketing 16(1), 49-63.

Silverman, D. 2011. Interpreting qualitative data. A guide to the principles of qualitative research. 4th ed. Great Britain: SAGE Publications Ltd.

Simon, H. 1996. The sciences of the artificial. 3rd ed. USA: MIT Press.

Strong, D.M. and Volkoff, O. 2010. Understanding organization-enterprise system fit: A path to theorizing the information technology artifact. MIS Quarterly 34(4), 731-756.

Tike 2014. The official webpage of Tike (The Information Centre of Ministry of Agriculture and Forestry). Retrieved from http://www.mmmtike.fi/. Accessed 15.5.2014.

Tuli, K., Kohli A., and Bharadwaj, S. 2007. Rethinking customer solutions: from product bundles to relational processes. Journal of Marketing vol. 71 (July 2007), 1–17.

Tuunanen, T., Myers, M., and Cassab, H. 2010. A conceptual framework for consumer information systems development. Pacific Asia Journal of the Association for Information Systems, 2, 47-66.

Vargo, S. and Lusch, R. 2004. Evolving to a new dominant logic for marketing. Journal of Marketing, 68 (January), 1–17.

Vartiainen, T., and Tuunanen, T. 2013. Co-creation of value for IT-enabled services: a case of geocaching. Proceedings of the 46th Hawaii International Conference on System Sciences, 1093-1101.

Väänänen-Vainio-Mattila, K. 2011. Käytettävyys ja käyttäjäkeskeinen suunnittelu. In A. Oulasvirta (Ed.), Ihmisen ja tietokoneen vuorovaikutus, pp. 102-126. Tallinna: Gaudeamus.

Walls, J., Widmeyer, G., and El Sawy, O. 1992. Building an information system design theory for vigilant EIS, information systems research, 3(1), 36-59. . Information Systems Research, 3(1), 36-59.

Walls, J., Widmeyer, G., and El Sawy, O. 2004. Assessing information system design theory in perspective: how useful was our 1992 initial rendition? Journal of Information Technology Theory and Application, 6(2), 43-58.

Walsham, G. 2006. Doing interpretive research. European Journal of Information Systems, (15), 320-330.

Weber, R. 1987. Toward a theory of artifacts: A paradigmatic base for information systems research. Journal of Information Systems, 1(1), 3-19.

Wynn, D. J., and Williams, C. K. 2012. Principles for conducting critical realist case study research in information systems. MIS Quarterly, 36(3), 787-810.

Yin, R. K. 2014. Case study research : Design and methods (5th ed. ed.). Los Angeles: SAGE.

Zhang, P., Scialdone, M., & Ku, M. (2011). IT artifacts and the state of IS research. International Conference on Information Systems 2011, Shanghai, China.

Unpublished references

Eteläaho, E. 2014. Case study: The utilization of Integration Readiness Level in Finnish national-level agricultural administration - case Tike. Unpublished manuscript.

Eteläaho, E., Tuomi, P., and Pirinen, R. 2015b. Improvement suggestions for the DSRM model. First version of the article for AMCIS. Unpublished manuscript.

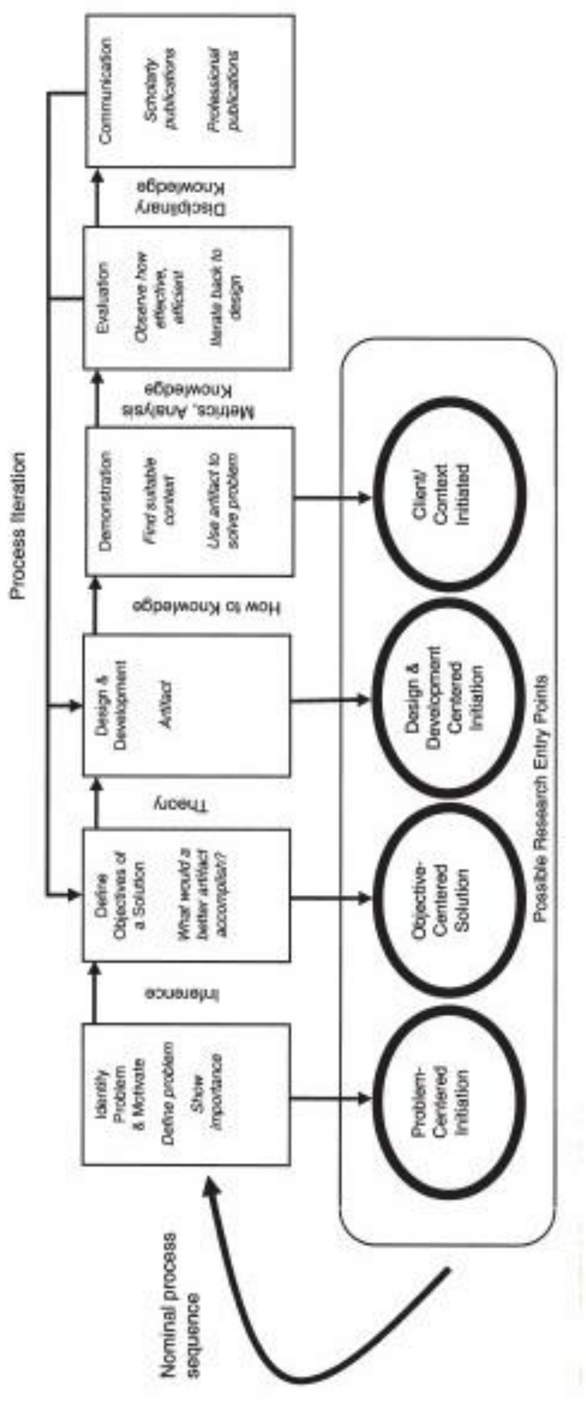
Tuunanen, T. and Peffers, K. 2011. Targeted Service Co-design Theory. Working draft. The University of Nevada Las Vegas.

Figures

Tables

Appendices

Appendix 1. Design Science Research Model by Peffers et al. (2008)



Appendix 2. Research attributes.

Title of study	Studies on DSRM model / Improvement suggestions for DSRM model (Peffers et al. 2008)
Research questions	How the DSRM model/framework can be modified to help in accomplishing better information systems?
Research agreement	
Unit of analysis	the experience of business information technology Master's degree students (n=12) of information systems projects; information system projects (n=10) and their documentation (n=40); literature on DSR (n=28);
Importance of study	Contribution to the development of the DSRM framework
Methodological focus	Single case study analysis; triangulation with another case study, DSR
Form of analysis	Mainly a qualitative analysis, saturation and triangulation.
Nature of study	Explanatory study of DSR in systems development projects.
Research Approach	Deductive investigation of DSRM information systems.
Specification of constructs	Design Science Research; information systems development
Theoretical approaches	Design Science Research; information systems research; Peffers et al. (2008) DSRM framework improvement
Theoretical literature	References of Design Science Research and Information Systems: Peffers et al. 2008; Hevner et al. 2004; Hevner and Chatterjee 2010; Peffers et al. 2006; March and Smith 1995; Walls et al. 1992; Simon 1996; Nunamaker et al. 1991; Gregor 2006; Gregor and Jones 2007; Gregor and Hevner 2013; Iivari 2007; Lyytinen and King 2004; Nunamaker and Briggs 2011; Orlikowski and Iacono 2001; Baskerville and Myers 2002; Avison and Elliot 2006; Weber 1987; Abdel-Hamid and Madnick 1989; Benbasat and Zmud 2003; Zhang et al. 2011; Grover and Lyytinen 2015; Germonprez et al. 2011; Pirinen 2015; Baskerville and Myers 2009; Walls et al. 2004; Markus et al. 2002; Alter 2003.
First research target	adding DSRM to projects (n=10) based on project documentation (n=40)
Outcome validation	Peffers, K., Tuunanen, T., Rothenberger, M., and Chatterjee, S. 2008. A design science research methodology for information systems research. Journal of Management Information Systems, 24(3), 45-77.
Research design	Single case design which includes the projects of Tike, the outcomes are compared to the reference (Peffers et al. 2008) and study unit data. Finally, the outcomes are compared to other organization's data and then concluded.
Logic of evidence	Replication logic: mainly literal replication logic.
Data analysis literature	Miles et al. 2014; Miles and Huberman 1994; Denzin and Lincoln 1994; Yin 2014; Silverman 2009; Robson 2011; Corbin and Strauss 2013; Campbell and Fiske 1959; Eisenhardt 1989; Gray 2009; Gerring 2007; Benbasat et al. 2002; Walsham 2006; Wynn and Williams 2012.
Data collection methods	Documents from work projects (n=40). The research data was coded, reduced and archived.
Questionnaire	
Coding	Each document was read carefully and then similar or identical problems or success factors were colored with certain color.
Notes	Researcher used notes from information system projects and made notes of thoughts arising from documents.
Team-based research	The conference article was written based on experiences in two organizations that were examined by two observers
Role description	Researchers as outsiders (objective) and as a participants in some projects as insiders (subjective).
Research consortium	
Research associations	Association for Information Systems (AIS); Association for Computing Machinery (ACM); and Institute of Electrical and Electronics Engineers (IEEE).